EAST Search History

Search #

Ref #	Hits	Search Query	DBs	Default Operator	Plurals	Time Stamp
L1	135	intel.asn. and register same (spill$3 or color$3 or allocat$4 or assign$4 or defin$5) same (symbolic$3 or pseudo or pseudo$1register or fake or temporar$4)	US-PGPUB; USPAT; USOCR; FPRS; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2007/04/18 16:32

doi.ieeecomputersociety.org/10.1109/TPDS.2007.10 - Similar pages

COS 320, Spring 2000. Programming **Assignment**
To get a new virtual **register** that has not been used before use the function Mips.newReg. Do not store any **temporaries** in real registers or on the stack. ...
www.cs.princeton.edu/courses/archive/spring05/cos320/**assignments**/a6.htm - 12k - Cached - Similar pages

Result Page: 1 2 3 4 5 6 7 8 9 10 **Next**

Terms used **register symbolic** Found **24,038** of **199,986**

Sort results by: relevance
Display results: expanded form
Save results to a Binder
Search Tips
Open results in a new window

Try an Advanced Search
Try this search in The ACM Guide

Results 1 - 20 of 200 Result page: **1** 2 3 4 5 6 7 8 9 10 next
Best 200 shown Relevance scale

1 Evicted variables and the interaction of global register allocation and symbolic debugging
Ali-Reza Adl-Tabatabai, Thomas Gross
March 1993 **Proceedings of the 20th ACM SIGPLAN-SIGACT symposium on Principles of programming languages POPL '93**
Publisher: ACM Press
Full text available: pdf(1.33 MB) Additional Information: full citation, abstract, references, citings, index terms

A symbolic debugger allows a user to display the values of program variables at a breakpoint. However, problems arise if the program is translated by an optimizing compiler. This paper addresses the effects of global register allocation and assignment: a register assigned to a variable V may not be holding V's value at a breakpoint since the register can also be assigned to other variables. We define the problem of determining whether a variable is in its assigned register as the re ...

2 Compilation and run-time systems: A faster optimal register allocator
Changqing Fu, Kent Wilken
November 2002 **Proceedings of the 35th annual ACM/IEEE international symposium on Microarchitecture MICRO 35**
Publisher: IEEE Computer Society Press
Full text available: pdf(982.37 KB) Publisher Site Additional Information: full citation, abstract, references, citings, index terms

Recently researchers have proposed modeling register allocation as an integer linear programming (IP) problem and solving it optimally for general purpose processors [17, 20] and for dedicated embedded systems [23]. Compared with traditional graph-coloring approaches, the IP-based allocators can improve a program's performance. However, the solution times are much slower.This paper presents an IP-based optimal register allocator which is much faster than previous work. We present several local a ...

3 Register allocation for irregular architectures
Bernhard Scholz, Erik Eckstein
June 2002 **ACM SIGPLAN Notices , Proceedings of the joint conference on Languages, compilers and tools for embedded systems: software and compilers for embedded systems LCTES/SCOPES '02**, Volume 37 Issue 7
Publisher: ACM Press
Additional Information: full citation, abstract, references, citings, index

Full text available: pdf(220.34 KB) terms

For irregular architectures global register allocation is still a challenging problem that has not been successfully solved so far. The graph-coloring analogy of traditional approaches does not match the needs of register allocation for such architectures which feature non-orthogonal instruction sets and small register files. This work proposes a fundamentally new approach to global register allocation for irregular architectures. Our approach formulates global allocation as a *partitioned boo ...*

Keywords: *boolean quadratic problem, register allocation*

4 Precise register allocation for irregular architectures

Timothy Kong, Kent D. Wilken

November 1998 **Proceedings of the 31st annual ACM/IEEE international symposium on Microarchitecture MICRO 31**

Publisher: IEEE Computer Society Press

Full text available: pdf(1.25 MB) Additional Information: full citation, references, citings, index terms

5 Formal verification of PowerPC arrays using symbolic trajectory evaluation

Manish Pandey, Richard Raimi, Derek L. Beatty, Randal E. Bryant

June 1996 **Proceedings of the 33rd annual conference on Design automation DAC '96**

Publisher: ACM Press

Full text available: pdf(122.46 KB) Additional Information: full citation, references, citings, index terms

6 Post-compaction register assignment in a retargetable compiler

Philip Sweany, Steven Beaty

November 1990 **Proceedings of the 23rd annual workshop and symposium on Microprogramming and microarchitecture MICRO 23**

Publisher: IEEE Computer Society Press

Full text available: pdf(998.12 KB) Additional Information: full citation, abstract, references, citings

We discuss graph-coloring register assignment in a retargetable compiler for Long-Instruction-Word architectures. Of specific concern is when, during the compilation process, should register assignment be performed. We conclude that, for best results, register assignment should follow compaction. We discuss methods of circumventing the implementation problems inherent in such late register assignment.

7 Register allocation with instruction scheduling

Shlomit S. Pinter

June 1993 **ACM SIGPLAN Notices , Proceedings of the ACM SIGPLAN 1993 conference on Programming language design and implementation PLDI '93**, Volume 28 Issue 6

Publisher: ACM Press

Full text available: pdf(931.91 KB) Additional Information: full citation, abstract, references, citings, index terms

We present a new framework in which considerations of both register allocation and instruction scheduling can be applied uniformly and simultaneously. In this framework an optimal coloring of a graph, called the parallel interference graph, provides an optimal register allocation and preserves the property that no false dependences are introduced, thus all the options for parallelism are kept for the scheduler to handle. For this framework we provide heuristics for trading ...

8 Symbolic simulation for correct machine design
William C. Carter, William H. Joyner, Daniel Brand
June 1979 **Proceedings of the 16th Conference on Design automation DAC '79**
Publisher: IEEE Press
Full text available: pdf(539.75 KB) Additional Information: full citation, abstract, citings, index terms

Program verification techniques which manipulate symbolic rather than actual values have been used successfully to find errors in implementations of computer designs. This paper describes symbolic simulation, a method similar to symbolic execution of programs, and its use in proving the correctness of machine architectures implemented in microcode. The procedure requires formal descriptions of machines at both the architectural and register transfer levels, but has been used to detect error ...

9 Partitioned register file for TTAs
Johan Janssen, Henk Corporaal
December 1995 **Proceedings of the 28th annual international symposium on Microarchitecture MICRO 28**
Publisher: IEEE Computer Society Press
Full text available: pdf(921.87 KB) Additional Information: full citation, references, citings, index terms

10 Formal verification: Handling special constructs in symbolic simulation
Alfred Kölbi, James Kukula, Kurt Antreich, Robert Damiano
June 2002 **Proceedings of the 39th conference on Design automation DAC '02**
Publisher: ACM Press
Full text available: pdf(109.20 KB) Additional Information: full citation, abstract, references, index terms

Symbolic simulation is a formal verification technique which combines the flexibility of conventional simulation with powerful symbolic methods. Some constructs, however, which are easy to handle in conventional simulation need special consideration in symbolic simulation. This paper discusses some special constructs that require unique treatment in symbolic simulation such as the symbolic representation of arrays, an efficient This paper discusses some special constructs that are unique to symb ...

Keywords: formal verification, symbolic simulation

11 Register allocation & spilling via graph coloring
G. J. Chaitin
June 1982 **ACM SIGPLAN Notices , Proceedings of the 1982 SIGPLAN symposium on Compiler construction SIGPLAN '82**, Volume 17 Issue 6
Publisher: ACM Press
Full text available: pdf(481.67 KB) Additional Information: full citation, abstract, references, citings, index terms

In a previous paper we reported the successful use of graph coloring techniques for doing global register allocation in an experimental PL/I optimizing compiler. When the compiler cannot color the register conflict graph with a number of colors equal to the number of available machine registers, it must add code to spill and reload registers to and from storage. Previously the compiler produced spill code whose quality sometimes left much to be desired, and the ad hoe techniques used took c ...

12 Formal hardware verification by symbolic ternary trajectory evaluation
Randal E. Bryant, Derek L. Beatty, Carl-Johan H. Seger
June 1991 **Proceedings of the 28th conference on ACM/IEEE design automation DAC '91**

Publisher: ACM Press
Full text available: pdf(613.01 KB) Additional Information: full citation, references, citings, index terms

13 Memory bank and register allocation in software synthesis for ASIPs
Ashok Sudarsanam, Sharad Malik
December 1995 **Proceedings of the 1995 IEEE/ACM international conference on Computer-aided design ICCAD '95**
Publisher: IEEE Computer Society
Full text available: pdf(82.67 KB) Publisher Site Additional Information: full citation, abstract, references, citings, index terms

An architectural feature commonly found in digital signal processors (DSPs) is multiple data-memory banks. This feature increases memory bandwidth by permitting multiple memory accesses to occur in parallel when the referenced variables belong to different memory banks and the registers involved are allocated according to a strict set of conditions. Unfortunately, current compiler technology is unable to take advantage of the potential increase in parallelism offered by such architectures. Conse ...

14 Philco/IBM translation at problem-oriented, symbolic and binary levels
Thomas M. Olsen
December 1965 **Communications of the ACM**, Volume 8 Issue 12
Publisher: ACM Press
Full text available: pdf(947.25 KB) Additional Information: full citation, references, index terms

15 The performance enhancement of descriptor-based virtual memory systems through the use of associative registers
R. E. Brundage, A. P. Batson
December 1974 **ACM SIGARCH Computer Architecture News , Proceedings of the 2nd annual symposium on Computer architecture ISCA '75**, Volume 3 Issue 4
Publisher: ACM Press
Full text available: pdf(539.04 KB) Additional Information: full citation, abstract, references, citings

Contemporary paged virtual memory systems often use associative registers to reduce access time to frequently-referenced pages. Here we examine the analogous use of associative registers in descriptorbased, symbolically-segmented virtual memory systems, where each segment contains an entire data structure as defined in a high-level language. Symbolic trace data from production Algol 60 programs were used to determine performance improvement as a function of the number of associative registers in ...

16 High level and architectural synthesis: A symbolic approach for the combined solution of scheduling and allocation
Gianpiero Cabodi, Mihai Lazarescu, Luciano Lavagno, Sergio Nocco, Claudio Passerone, Stefano Quer
October 2002 **Proceedings of the 15th international symposium on System Synthesis ISSS '02**
Publisher: ACM Press
Full text available: pdf(132.26 KB) Additional Information: full citation, abstract, references, index terms

Scheduling is widely recognized as a very important step in high-level synthesis. Nevertheless, it is usually done without taking into account the effects on the actual hardware implementation. This paper presents an efficient symbolic technique to concurrently integrate operation scheduling and resource allocation. The technique inherits all the features of "standard" BDD-based control dominated scheduling, including

resource-constraining, speculation and pruning. In addition, it introduces an ...

Keywords: BDD, allocation, automata, high-level synthesis, scheduling

17 Symbolic simulation for functional verification with ADLIB and SDL

W. E. Cory

June 1981 **Proceedings of the 18th conference on Design automation DAC '81**

Publisher: IEEE Press

Full text available: pdf(618.02 KB) Additional Information: full citation, abstract, references, citings, index terms

The basic verification problem addressed in this paper is to determine the consistency of two digital design descriptions. This is done by symbolically simulating each description and comparing the results. This approach is complicated by the presence of different levels of abstraction and asynchronous timing. This paper motivates interest in this problem and provides background information on verification, ADLIB, and SDL. It then discusses approaches for dealing with the problem ...

18 Symbolic modeling and evaluation of data paths

Chuck Monahan, Forrest Brewer

January 1995 **Proceedings of the 32nd ACM/IEEE conference on Design automation DAC '95**

Publisher: ACM Press

Full text available: pdf(73.94 KB) Additional Information: full citation, references, citings, index terms

19 A novel framework of register allocation for software pipelining

Qi Ning, Guang R. Gao

March 1993 **Proceedings of the 20th ACM SIGPLAN-SIGACT symposium on Principles of programming languages POPL '93**

Publisher: ACM Press

Full text available: pdf(1.30 MB) Additional Information: full citation, abstract, references, citings, index terms

Although software pipelining has been proposed as one of the most important loop scheduling methods, simultaneous scheduling and register allocation is less understood and remains an open problem [28]. The objective of this paper is to develop a unified algorithmic framework for concurrent scheduling and register allocation to support time-optimal software pipelining. A key intuition leading to this surprisingly simple formulation and its efficient solution is the association of maximum com ...

20 Energy-aware systems: Binary translation to improve energy efficiency through post-pass register re-allocation

Kun Zhang, Tao Zhang, Santosh Pande

September 2004 **Proceedings of the 4th ACM international conference on Embedded software EMSOFT '04**

Publisher: ACM Press

Full text available: pdf(190.98 KB) Additional Information: full citation, abstract, references, citings, index terms

Energy efficiency is rapidly becoming a first class optimization parameter for modern systems. Caches are critical to the overall performance and thus, modern processors (both high and low-end) tend to deploy a cache with large size and high degree of associativity. Due a large size cache power takes up a significant percentage of total system power. One important way to reduce cache power consumption is to reduce the

dynamic activities in the cache by reducing the dynamic load-store counts. In ...

Keywords: cache power consumption, dead registers, register re-allocation, unused registers

Results 1 - 20 of 200 Result page: **1** 2 3 4 5 6 7 8 9 10 next

Web Results **1** - **10** of about **505,000** for **register assignment (symbolic or temporaries)**. (**0.17** seconds)

[PDF] Post-compaction **register assignment** in a retargetable compiler ...
File Format: PDF/Adobe Acrobat
edge that a later **register assignment** will "make every-. thing right" with respect to the ...
Isym14 are **temporaries**. and the following **symbolic register** ...
ieeexplore.ieee.org/iel2/327/3975/00151432.pdf?arnumber=151432 - Similar pages

[PDF] Post-Compaction **Register Assignment** in a Retargetable Compiler ...
File Format: PDF/Adobe Acrobat
edge that a later **register assignment** will "make every- ... Isyml3 and Isyml4 are **temporaries**. and the following. **symbolic register**. mappings have been made ...
portal.acm.org/ft_gateway.cfm?id=255257&type=pdf - Similar pages

> [PDF] Evicted Variables and the Interaction of Global **Register** ...
> File Format: PDF/Adobe Acrobat
> **Register**. r3. is also used as an expression **temporary**. at instruction 13. No code has been generated to perform. a store to a for S1, since S1's **assignment** ...
> portal.acm.org/ft_gateway.cfm?id=158692&type=pdf - Similar pages
> [More results from portal.acm.org]

PATCH: **symbolic** regnames for sysdeps/powerpc/stpcpy.S
-EALIGN(__stpcpy,4,0) /* char * [r3] stpcpy (char *dest [r3], const char *src [r4]) */ -/* General **register assignments**: - r0: **temporary** - r3: pointer to ...
sourceware.org/ml/libc-hacker/2000-06/msg00282.html - 8k - Cached - Similar pages

PATCH: **symbolic** regnames for sysdeps/powerpc/{str,mem}*.S
-EALIGN(strcpy,4,0) /* char * [r3] strcpy (char *dest [r3], const char *src [r4]) */ -/* General **register assignments**: - r0: **temporary** - r3: saved `dest' ...
sources.redhat.com/ml/libc-hacker/2000-06/msg00017.html - 28k - Cached - Similar pages

> sysdeps/powerpc/*.S, BPs and registers
> ... since BPs don't alter the **register assignments** for arguments (a bounded ... r4 #endif All **temporary register** variables will also need **symbolic** names. ...
> sources.redhat.com/ml/libc-hacker/2000-05/msg00072.html - 6k - Cached - Similar pages

[PDF] THESIS **REGISTER** ALLOCATION AND **ASSIGNMENT** IN A RETARGETABLE ...
File Format: PDF/Adobe Acrobat - View as HTML
of compilation to represent the program variables and compiler-generated **temporaries**. A. unique name usually represents each **symbolic register**, generally an ...
emess.mscd.edu/~beaty/Dossier/Papers/thesis.pdf - Similar pages

[PDF] LaTTe: A Java VM Just-In-Time Compiler with Fast and Efficient ...
File Format: PDF/Adobe Acrobat - View as HTML
location: s = operand stack, l = local variable, t = generated **temporary**. • number further distinguishing the **symbolic register** ...
www.complang.tuwien.ac.at/anton/lvas/sem06w/molnar.pdf - Similar pages

Efficient **Register** Mapping and Allocation in LaTTe, an Open-Source ...
It maps multiple **symbolic** registers to the same real **register** when they are equal, and uses clever heuristics to match physical **register assignments** across ...